

WHAT IS CLAIMED IS:

1. A method for processing content requests in a network having at least one content provider having stored documents in an unprocessed format, a proxy server and at least one client, the method comprising the following steps performed by the proxy server, of:
receiving a request from the client for a document associated with the content

provider;

determining whether a processed version of the document is located in a local cache;

when it is determined that the processed version of the document is located in the local cache, providing the processed version of the document to the client; and

when it is determined that the processed version of the document is not located in the local cache, (i) obtaining the unprocessed document associated with the content provider; (ii) processing the unprocessed document in accordance with predetermined instructions associated with the unprocessed document; and (iii) providing the processed version of the identified processed document to the client.

2. The method of claim 1, wherein the step of processing the unprocessed document, further comprises the step of storing the processed version of the document in the local cache.

3. The method of claim 1, wherein the step of determining whether a processed version of the document is located in local cache, further comprises the step of

5 ascertaining whether the document is written in a second markup language, wherein said second markup language is a processed version of said first markup language.

10 4. The method of claim 1, wherein the step of processing the unprocessed document in accordance with predetermined instructions associated with the unprocessed document, further comprises the step of converting the unprocessed document from a first markup language to a second markup language.

15 5. The method of claim 1, wherein said second markup language is the extensible markup language (XML).

20 6. A method for processing content requests in a network having at least one content provider having stored documents in an unprocessed format, a proxy server and at least one client, the method comprising the following steps performed by the proxy server, of:

 receiving an unprocessed document from the content provider, wherein the document was requested by the client;

 determining whether a processed version of the document is located in a local cache;

 when it is determined that the processed version of the document is located in the local cache, providing the processed version of the document to the client; and

25 when it is determined that the processed version of the document is not located in the local cache, (i) processing the unprocessed document in accordance with predetermined

5 instructions associated with the unprocessed document; and (ii) providing the processed version of the identified processed document to the client.

7. The method of claim 6, wherein the step of processing the unprocessed document, further comprises the step of storing the processed version of the document in the
10 local cache.

8. The method of claim 6, wherein the step of determining whether a processed version of the document is located in local cache, further comprises the step of
15 ascertaining whether the document is written in a second markup language, wherein said second markup language is a processed version of said first markup language.

9. The method of claim 6, wherein the step of processing the unprocessed document in accordance with predetermined instructions associated with the unprocessed document, further comprises the step of converting the unprocessed document from a first
20 markup language to a second markup language.

10. An Internet Proxy server comprising:
a memory having program instructions; and
a processor configured to use the program instructions to receive a request
25 from the client for a document associated with the content provider; to determine whether a processed version of the document is located in a local cache; when it is determined that the

5 processed version of the document is located in the local cache, to provide the processed
version of the document to the client; and when it is determined that the processed version of
the document is not located in the local cache, (i) obtain the unprocessed document associated
with the content provider; (ii) process the unprocessed document in accordance with
predetermined instructions associated with the unprocessed document; and (iii) provide the
10 processed version of the identified processed document to the client.

11. The server of claim 10, wherein the instruction to process the
unprocessed document, further comprises the step of storing the processed version of the
document in the local cache.

12. The server of claim 10, wherein the instruction to determine whether a
processed version of the document is located in local cache, further comprises the instruction
to ascertain whether the document is written in a second markup language, wherein said
second markup language is a processed version of said first markup language.

13. The server of claim 10, wherein the instruction to process the
unprocessed document in accordance with predetermined instructions associated with the
unprocessed document, further comprises the instruction to convert the unprocessed document
from a first markup language to a second markup language.

5

14. The server of claim 10, wherein said second markup language is the extensible markup language (XML).

15. An Internet Proxy server comprising:

a memory having program instructions; and

10

a processor configured to use the program instructions to receive an unprocessed document from the content provider, wherein the document was requested by the client; to determine whether a processed version of the document is located in a local cache; when it is determined that the processed version of the document is located in the local cache, to provide the processed version of the document to the client; and when it is determined that the processed version of the document is not located in the local cache, (i) process the unprocessed document in accordance with predetermined instructions associated with the unprocessed document; and (ii) provide the processed version of the identified processed document to the client.

15

20

16. The server of claim 15, wherein the instruction to process the unprocessed document, further comprises the instruction of storing the processed version of the document in the local cache.

25

17. The server of claim 15, wherein the instruction to determine whether a processed version of the document is located in local cache, further comprises the step to

5 ascertain whether the document is written in a second markup language, wherein said second markup language is a processed version of said first markup language.

18. The method of claim 15, wherein the instruction to process the unprocessed document in accordance with predetermined instructions associated with the unprocessed document, further comprises the instruction to convert the unprocessed document from a first markup language to a second markup language.

19. A data processing system for processing content requests in a network having at least one content provider having stored documents in an unprocessed format, a proxy server and at least one client, the system comprising:

means for receiving a request from the client for a document associated with the content provider;

means for determining whether a processed version of the document is located in a local cache;

when it is determined that the processed version of the document is located in the local cache, means for providing the processed version of the document to the client; and

when it is determined that the processed version of the document is not located in the local cache, (i) means for obtaining the unprocessed document associated with the content provider; (ii) means for processing the unprocessed document in accordance with predetermined instructions associated with the unprocessed document; and (iii) means for providing the processed version of the identified processed document to the client.